

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-024974
(43)Date of publication of application : 29.01.1999

(51)Int.Cl. G06F 12/00

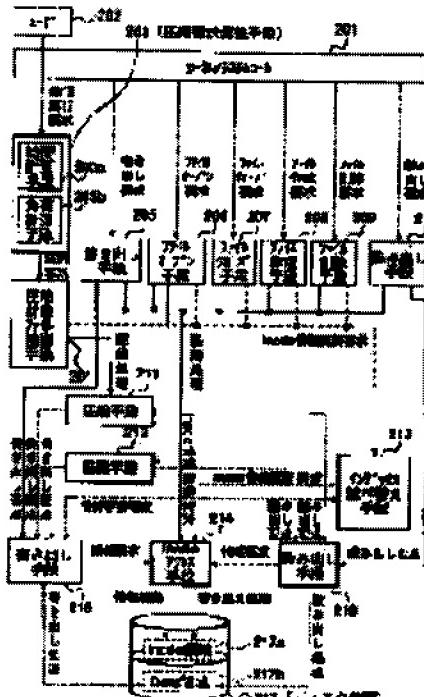
(21)Application number : 09-196431 (71)Applicant : CANON INC
(22)Date of filing : 08.07.1997 (72)Inventor : HIRAHARA ATSUSHI
SASAKI SEIJI
WATANABE HIROYASU
BESSHO MASATAKA

(54) FILE MANAGING DEVICE/METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a file managing device/method for controlling timewise overhead on a CPU load in compression/extension, by taking a method for improving the availability of a disk by means of compressing individual files in accordance with the access frequency of the file, and of preventing frequently accessed files from being compressed.

SOLUTION: This managing device has an index rearranging means 213 for rearranging the order of the indexes of files with the leading address as a starting address when the file is accessed, a file extraction means for compression object 204, which extracts the file whose final access time is the oldest from the indexes of the files arranged in a final access order and a compression request issuing means 203 for determining whether a compression processing is executed or is postponed by considering the loading situation of the system.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision]

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平11-24974

(43)公開日 平成11年(1999)1月29日

(51)Int.Cl.⁶

G 0 6 F 12/00

識別記号

5 2 0

F I

G 0 6 F 12/00

5 2 0 A

審査請求 未請求 請求項の数14 FD (全 20 頁)

(21)出願番号 特願平9-196431

(22)出願日 平成9年(1997)7月8日

(71)出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72)発明者 平原 厚志

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 佐々木 誠司

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 渡辺 浩廉

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(74)代理人 弁理士 渡部 敏彦

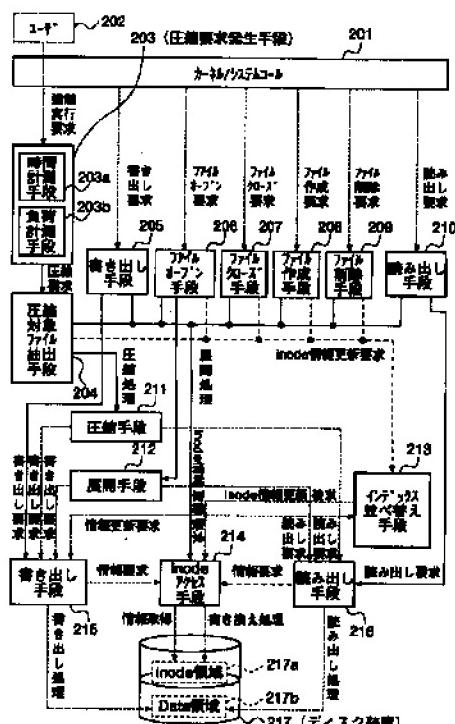
最終頁に続く

(54)【発明の名称】 ファイル管理装置及びファイル管理方法

(57)【要約】

【課題】 ファイルのアクセス頻度に応じて個々のファイルを圧縮することにより、ディスクの使用効率を向上させると共に、頻繁にアクセスされるファイルに関しては圧縮しないという方法をとることにより、圧縮・展開にかかる時間的、C P U負荷的なオーバヘッドを抑制すること等を可能としたファイル管理装置及びファイル管理方法を提供する。

【解決手段】 ファイルのアクセス発生時に当該ファイルのインデックスの順番を先頭に並べ替えるインデックス並べ替え手段2 1 3と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する圧縮対象ファイル抽出手段2 0 4と、システムの負荷状況を考慮して圧縮処理を実行するか延期するかを決定する圧縮要求発生手段2 0 3とを有する。



【特許請求の範囲】

【請求項1】 ファイルを蓄積し管理するファイル管理装置であって、

ファイルのアクセス発生時に当該ファイルの実体の位置を示すインデックスの順番を先頭に並べ替える並べ替え手段と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する抽出手段と、抽出したファイルの状態を条件に応じて変更する変更手段とを有することを特徴とするファイル管理装置。

【請求項2】 ファイルが利用されるシステムの負荷状況を計測する負荷計測手段と、時刻を計測する時間計測手段とを有し、前記変更手段は、ファイルの状態変更をシステムの負荷が軽い時間に行なうことを特徴とする請求項1記載のファイル管理装置。

【請求項3】 前記ファイルの状態とは、ファイルがオープンされた使用中、ファイルが使用中でもなく圧縮もされていない通常状態、ファイルが圧縮された圧縮状態、ファイルが利用されていない未使用状態であることを特徴とする請求項1記載のファイル管理装置。

【請求項4】 前記並べ替え手段は、通常状態のファイルにオープン要求があった場合はそのインデックスを使用中列の先頭に移動させ、使用中のファイルにクローズ要求があった場合はそのインデックスを通常状態列の先頭に移動させ、通常状態のファイルに圧縮要求があった場合はそのインデックスを圧縮状態列の先頭に移動させ、圧縮状態のファイルにアクセス要求があった場合はそのインデックスを使用中列の先頭に移動させ、通常状態或いは圧縮状態のファイルに削除要求があった場合はそのインデックスを未使用状態列の先頭に移動させることを特徴とする請求項1記載のファイル管理装置。

【請求項5】 前記変更手段は、最終アクセス時刻、ファイルの種類等の条件に応じてファイルを圧縮することを特徴とする請求項1記載のファイル管理装置。

【請求項6】 各ファイルのインデックス列は双向リンクリストを用いることを特徴とする請求項1記載のファイル管理装置。

【請求項7】 各ファイルのインデックス列は円形配列を用いることを特徴とする請求項1記載のファイル管理装置。

【請求項8】 ファイルを蓄積し管理するファイル管理办法であって、

ファイルのアクセス発生時に当該ファイルの実体の位置を示すインデックスの順番を先頭に並べ替える並べ替え手段と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する抽出手段と、抽出したファイルの状態を条件に応じて変更する変更手段とを有することを特徴とするファイル管理方法。

【請求項9】 ファイルが利用されるシステムの負荷状

況を計測する負荷計測ステップと、時刻を計測する時間計測ステップとを有し、前記変更ステップでは、ファイルの状態変更をシステムの負荷が軽い時間に行なうことを行なうことを特徴とする請求項8記載のファイル管理方法。

【請求項10】 前記ファイルの状態とは、ファイルがオープンされた使用中、ファイルが使用中でもなく圧縮もされていない通常状態、ファイルが圧縮された圧縮状態、ファイルが利用されていない未使用状態であることを特徴とする請求項8記載のファイル管理方法。

【請求項11】 前記並べ替えステップでは、通常状態のファイルにオープン要求があった場合はそのインデックスを使用中列の先頭に移動させ、使用中のファイルにクローズ要求があった場合はそのインデックスを通常状態列の先頭に移動させ、通常状態のファイルに圧縮要求があった場合はそのインデックスを圧縮状態列の先頭に移動させ、圧縮状態のファイルにアクセス要求があった場合はそのインデックスを使用中列の先頭に移動させ、通常状態或いは圧縮状態のファイルに削除要求があった場合はそのインデックスを未使用状態列の先頭に移動させることを特徴とする請求項8記載のファイル管理方法。

【請求項12】 前記変更ステップでは、最終アクセス時刻、ファイルの種類等の条件に応じてファイルを圧縮することを特徴とする請求項8記載のファイル管理方法。

【請求項13】 各ファイルのインデックス列は双向リンクリストを用いることを特徴とする請求項8記載のファイル管理方法。

【請求項14】 各ファイルのインデックス列は円形配列を用いることを特徴とする請求項8記載のファイル管理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ファイル管理装置及びファイル管理办法に係り、更に詳しくは、コンピュータシステムにおいて多数のファイルを管理するハードディスク等の外部記憶装置で、使用頻度等の条件によりファイルを圧縮・非圧縮等の状態に変更してディスクのスペースを有効利用する場合に好適なファイル管理装置及びファイル管理办法に関する。

【0002】

【従来の技術】 従来、パーソナルコンピュータやワークステーション等のコンピュータに装備されているディスク装置(ファイルシステム)上では、多数のファイルを管理・蓄積することができる。ところが、ディスクを使用するに伴いファイル数及び使用容量は増加し、殆どアクセスされることのないファイル等でディスクのスペースを圧迫する。従って、ディスクスペースの無駄を省く何等かの工夫が必要となる。

【0003】 ファイルシステムのスペースの無駄を省く

方式(以下、省スペース方式)としては、ファイルの圧縮技術がある。現在、以下のようなファイル圧縮技術を利用することができます。

【0004】(1) 特定のファイルに対して、ユーザが圧縮プログラムを適用してファイルサイズを小さくする。そのファイルを利用する場合は、展開プログラムを用いて元のファイルを復元する。

【0005】(2) 全てのファイルに対して、ファイルをファイルシステムに書き出す時に圧縮操作を行い、ファイルを読み出す時に展開操作を行う。

【0006】(3) 例えば特開平6-75834号公報に記載された「ファイルスペース確保方式」では、圧縮を用いたファイルシステムの有効な利用法を示している。該ファイルシステム確保方式を用いた装置の特徴は、利用者が適切な未使用日数の閾値を設定することで、ディスク装置上のファイルスペースの空きが無くなつても、低アクセス頻度のファイルの内、指定された未使用日数の閾値以上アクセスされていないファイルのデータを圧縮して空きスペースを作成し、スペースを有効利用するものである。

【0007】

【発明が解決しようとする課題】しかしながら、上述したファイルシステムのスペースの無駄を省く従来の省スペース方式には、下記のような問題があった。

【0008】上記(1)の方法は、目的のファイルを使用するにあたり、それを扱うユーザがファイルの圧縮、展開作業を行う必要があり手間がかかる(課題1)。

【0009】上記(2)の方法は、上記課題1を解決しているが、全てのファイルに対して圧縮・展開操作を行うため、CPU負荷的なオーバヘッド(コンピュータの使用時にその仕事には直接関係無いが間接的には一見無駄に費やされるように見える時間)がかかる(課題2)。また、高速なファイルアクセスが要求されるようなアプリケーションにおいては、ファイルの圧縮・展開にかかる時間が障害になる場合がある(課題3)。

【0010】上記(3)の方法は、上記課題1、2を解決しているが、上記課題3を解決していない。即ち、「ファイルスペース確保方式」は、ファイルの書き込み・拡張要求があったときディスクに空き容量が無い場合に、ディスク上の全ファイルに対してアクセス頻度の情報を収集して、アクセス頻度の最も低いファイルを圧縮しファイルスペースを確保する方式である。そのため、一旦ディスクが満杯になると圧縮操作が頻発し、高速なファイルアクセスが不可能になるものと思われる。

【0011】本発明は、上述した点に鑑みなされたものであり、ファイルのアクセス頻度に応じて個々のファイルを圧縮することにより、ディスクの使用効率を向上させると共に、頻繁にアクセスされるファイルに関しては圧縮しないという方法をとることにより、圧縮・展開にかかる時間的、CPU負荷的なオーバヘッドを抑制する

こと等を可能としたファイル管理装置及びファイル管理办法を提供することを目的とする。

【0012】

【課題を解決するための手段】上記目的を達成するため、請求項1の発明は、ファイルを蓄積し管理するファイル管理装置であって、ファイルのアクセス発生時に当該ファイルの実体の位置を示すインデックスの順番を先頭に並べ替える並べ替え手段と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する抽出手段と、抽出したファイルの状態を条件に応じて変更する変更手段とを有することを特徴とする。

【0013】上記目的を達成するため、請求項2の発明は、ファイルが利用されるシステムの負荷状況を計測する負荷計測手段と、時刻を計測する時間計測手段とを有し、前記変更手段は、ファイルの状態変更をシステムの負荷が軽い時間に行うことを特徴とする。

【0014】上記目的を達成するため、請求項3の発明は、前記ファイルの状態とは、ファイルがオープンされた使用中、ファイルが使用中でもなく圧縮もされていない通常状態、ファイルが圧縮された圧縮状態、ファイルが利用されていない未使用状態であることを特徴とする。

【0015】上記目的を達成するため、請求項4の発明は、前記並べ替え手段は、通常状態のファイルにオープン要求があった場合はそのインデックスを使用中列の先頭に移動させ、使用中のファイルにクローズ要求があった場合はそのインデックスを通常状態列の先頭に移動させ、通常状態のファイルに圧縮要求があった場合はそのインデックスを圧縮状態列の先頭に移動させ、圧縮状態のファイルにアクセス要求があった場合はそのインデックスを使用中列の先頭に移動させ、通常状態或いは圧縮状態のファイルに削除要求があった場合はそのインデックスを未使用状態列の先頭に移動させることを特徴とする。

【0016】上記目的を達成するため、請求項5の発明は、前記変更手段は、最終アクセス時刻、ファイルの種類等の条件に応じてファイルを圧縮することを特徴とする。上記目的を達成するため、請求項6の発明は、各ファイルのインデックス列は双向リンクリストを用いることを特徴とする。

【0017】上記目的を達成するため、請求項7の発明は、各ファイルのインデックス列は円形配列を用いることを特徴とする。

【0018】上記目的を達成するため、請求項8の発明は、ファイルを蓄積し管理するファイル管理方法であって、ファイルのアクセス発生時に当該ファイルの実体の位置を示すインデックスの順番を先頭に並べ替える並べ替えステップと、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出す

る抽出ステップと、抽出したファイルの状態を条件に応じて変更する変更ステップとを有することを特徴とする。

【 0 0 1 9 】 上記目的を達成するため、請求項9の発明は、ファイルが利用されるシステムの負荷状況を計測する負荷計測ステップと、時刻を計測する時間計測ステップとを有し、前記変更ステップでは、ファイルの状態変更をシステムの負荷が軽い時間に行なうことを特徴とする。

【 0 0 2 0 】 上記目的を達成するため、請求項10の発明は、前記ファイルの状態とは、ファイルがオープンされた使用中、ファイルが使用中でもなく圧縮もされていない通常状態、ファイルが圧縮された圧縮状態、ファイルが利用されていない未使用状態であることを特徴とする。

【 0 0 2 1 】 上記目的を達成するため、請求項11の発明は、前記並べ替えステップでは、通常状態のファイルにオーブン要求があった場合はそのインデックスを使用中列の先頭に移動させ、使用中のファイルにクローズ要求があった場合はそのインデックスを通常状態列の先頭に移動させ、通常状態のファイルに圧縮要求があった場合はそのインデックスを圧縮状態列の先頭に移動させ、圧縮状態のファイルにアクセス要求があった場合はそのインデックスを使用中列の先頭に移動させ、通常状態或いは圧縮状態のファイルに削除要求があった場合はそのインデックスを未使用状態列の先頭に移動させることを特徴とする。

【 0 0 2 2 】 上記目的を達成するため、請求項12の発明は、前記変更ステップでは、最終アクセス時刻、ファイルの種類等の条件に応じてファイルを圧縮することを特徴とする。

【 0 0 2 3 】 上記目的を達成するため、請求項13の発明は、各ファイルのインデックス列は双向リンクリストを用いることを特徴とする。

【 0 0 2 4 】 上記目的を達成するため、請求項14の発明は、各ファイルのインデックス列は円形配列を用いることを特徴とする。

【 0 0 2 5 】

【 発明の実施の形態】 先ず最初に、本発明の実施の形態を説明する前に、本発明の原理について説明する。本発明に係る自動圧縮ファイルシステムは、例えば以下のような構成となる。即ち、本発明に係る自動圧縮ファイルシステムのファイルの状態は、(1) 「 使用中 」 、(2) 「 通常(展開) 状態 」 、(3) 「 圧縮状態 」 、(4) 「 未使用状態 」 の何れかの状態にある。

【 0 0 2 6 】 上記の各状態において、(1) 「 使用中 」 は、OS (Operating System) またはアプリケーション・プログラムによって使用中(ファイルオープン中) の状態で、この状態のファイルはクローズされるまで他の状態には遷移しない。(2) 「 通常状態 」 は、ファイ

ルは使用中ではなく、圧縮もされていない普通の状態である。(3) 「 圧縮状態 」 は、「 通常状態 」 のファイルに圧縮処理を施した状態である。このファイルを使用する場合は、ファイルの展開処理を施してから利用する。(4) 「 未使用状態 」 は、状態(1) 、(2) 、(3) 以外の状態で、利用されていない状態(データ領域を持たない状態) を示す。

【 0 0 2 7 】 そして、各ファイルの実体の位置を指すインデックス列は、双向リンクリストや配列等で実現されており、そのインデックス列の任意の位置のインデックスをインデックス列の先頭に並べ替えるインデックス並べ替え手段を備える。このインデックス列は、上述した「 ファイルの状態 」 別にそれぞれ別々のインデックス列として管理される。ファイルの状態が変化する度に、そのインデックス列を別のインデックス列の先頭に並べ替える手段を持つ。例えば、或るファイルが(2) 通常状態から(3) 圧縮状態に変化した場合、(2) 通常状態のインデックス列(通常状態列) にあるインデックスは、(3) 圧縮状態のインデックス列(圧縮状態列) の先頭に並べ替えられる。

【 0 0 2 8 】 また、次に示す方法でファイルの状態を変更する。即ち、

1 . 通常状態→使用中： 通常状態のファイルに対してファイルOpen要求があった場合は、そのファイルのインデックスを(1) 使用中列の先頭に移動させる。

【 0 0 2 9 】 2 . 使用中→通常状態： 使用中のファイルに対してファイルClose要求があった場合は、そのファイルのインデックスを(2) 通常状態列の先頭に移動させる。

【 0 0 3 0 】 3 . 通常状態→圧縮状態： 後述するファイル圧縮要求があった場合は、対象となるファイルを圧縮手段で圧縮し、(2) 通常状態列からインデックスを取り出し、(3) 圧縮状態列の先頭に移動させる。

【 0 0 3 1 】 4 . 圧縮状態→使用中： 圧縮状態にあるファイルに対してアクセス(Read/Write 等) が発生した場合は、展開手段で元の状態に展開され、インデックスを使用中列の先頭に移動させる。

【 0 0 3 2 】 5 . 通常状態、圧縮状態→未使用状態： ファイルの削除要求があった場合は、使用していたデータ領域を開放しファイルの状態は未使用状態に変更される。また、インデックスを(4) 未使用状態列の先頭に移動させる。

【 0 0 3 3 】 上記の手段を備えたファイルシステムにおいて、頻繁にアクセスされるファイルのインデックスは常に(2) 通常状態列の前方に位置し、アクセス頻度が低いファイルのインデックスは(2) 通常状態列の後方に移動していく。従って、係る自動圧縮ファイルシステムのインデックスの(2) 通常状態列は、常に最新のアクセス順に整列されていることになる。

【 0 0 3 4 】 圧縮処理は、圧縮対象ファイル抽出手段に

7

よって選ばれたファイルに対して行われる。該圧縮対象ファイル抽出手段は、上述した(2)通常状態列を末尾から順に検索してゆき、ファイルの属性(最終アクセス時刻、ファイルの種類、パーミッション：アクセス許可)を考慮して圧縮するか否かを決定する。例えば、検索したファイルの種類がデバイススペシャルファイル等の場合は圧縮しない。

【 0 0 3 5 】 圧縮対象ファイルが決定された場合は、ファイル圧縮処理命令を圧縮手段に対して発行する。また、該圧縮対象ファイル抽出の作業は、現在時刻とそのファイルの最終アクセス時刻との差が、本ファイルシステムで定める値(ユーザによって変更可能)よりも小さくなるまで検索を繰り返す。即ち、(2)通常状態列はアクセス順に整列されているので全検索する必要はない。

【 0 0 3 6 】 また、圧縮対象ファイル抽出手段は、圧縮要求発生手段によって発生される圧縮要求により動作する。圧縮要求発生手段は、ユーザからの直接要求で起動する場合と、時間計測手段によって一定時間毎に自動的に起動される場合がある。本ファイルシステムでは、ユーザからの圧縮処理開始要求があった場合は強制的に圧縮対象ファイル抽出手段による処理が実行される。通常は、一定時間毎にその時のシステムの負荷状況を負荷計測手段により獲得し、それを考慮して圧縮処理を実行するか延期するかを決定する。

【 0 0 3 7 】 以下、本発明の好適な第1及び第2の実施の形態について図面を参照して詳細に説明する。

【 0 0 3 8 】 [1] 第1の実施の形態

本発明の第1の実施の形態においては、例えばUNIX(米国AT&T社ベル研究所で開発されたタイムシェアリングシステム用オペレーティングシステム)のファイルシステムに適用した例を示す。第1の実施の形態では、上述したファイルのインデックスを本発明に適するよう拡張したi_nodeとして実現する。

【 0 0 3 9 】 即ち、各ファイルのi_nodeは、それぞれが双方向のリンクリストで繋がれており、そのi_nodeリストの任意の位置のi_nodeをi_nodeリスト並べ替え手段を備える。この手段は、上述したインデックス並べ替え手段に相当する。この手段を用いてファイルに対するアクセスが生じたときに、そのi_nodeのリンク順序をi_nodeリストの先頭に配置する。

【 0 0 4 0 】 このi_nodeのリンクリストは、上述した「ファイルの状態」別にそれぞれ別々のリストとして管理される。ファイルの状態が変化する度に、そのノードを別のリストの先頭に繋ぎかえる手段を持つ。例えば、或るファイルが(2)通常状態から(3)圧縮状態に変化した場合、(2)通常状態のリスト(通常状態リスト)に繋がっていたノードは、(3)圧縮状態のリスト(圧縮状態リスト)の先頭へ繋ぎかえられる。

8

【 0 0 4 1 】 また、次に示す方法でファイルの状態を変更する。即ち、

1. 通常状態→使用中：通常状態のファイルに対してファイルOpen要求があった場合は、そのファイルのi_nodeを(1) 使用中リストの先頭に移動させる。

【 0 0 4 2 】 2. 使用中→通常状態：使用中のファイルに対してファイルClose要求があった場合は、そのファイルのi_nodeを(2) 通常状態リストの先頭に移動させる。

10 【 0 0 4 3 】 3. 通常状態→圧縮状態：ファイル圧縮要求があった場合は、対象となるファイルを圧縮手段で圧縮し、(2) 通常状態リストからi_nodeを取り出し、(3) 圧縮状態リストの先頭に移動させる。

【 0 0 4 4 】 4. 圧縮状態→使用中：圧縮状態にあるファイルに対してアクセス(Read/Write等)が発生した場合は、展開手段で元の状態に展開され、i_nodeを使用中リストの先頭に移動させる。

20 【 0 0 4 5 】 5. 通常状態、圧縮状態→未使用状態：ファイルの削除要求があった場合は、使用していたデータ領域を開放しファイルの状態は未使用状態に変更される。また、管理情報ノードを(4) 未使用状態リストの先頭に移動させる。

【 0 0 4 6 】 上記の手段を備えたファイルシステムにおいて、頻繁にアクセスされるファイルのi_nodeは常に(2) 通常状態リストの前に位置し、アクセス頻度が低いファイルのi_nodeは(2) 通常状態リストの後方に移動していく。従って、係るファイルシステムのi_nodeの(2) 通常状態リストは、常に最新のアクセス順にリンクされていることになる。

30 【 0 0 4 7 】 圧縮処理は、圧縮対象ファイル抽出手段によって選ばれたファイルに対して行われる。該圧縮対象ファイル抽出手段は、上述した(2) 通常状態リストを末尾から順に辿ってゆき、ファイルの属性(最終アクセス時刻、ファイルの種類、パーミッション)を考慮して圧縮するか否かを決定する。例えば、検索したファイルの種類がデバイススペシャルファイルやシンボリックリンクファイル、ソケットファイル等の場合は圧縮しない。

40 【 0 0 4 8 】 圧縮対象ファイルが決定された場合は、ファイル圧縮処理命令を圧縮手段に対して発行する。また、該圧縮対象ファイル抽出の作業は、現在時刻とそのファイルの最終アクセス時刻との差が、本ファイルシステムで定める値(ユーザによって変更可能)よりも小さくなるまで検索を繰り返す。即ち、(2) 通常状態リストはアクセス順にリンクされているので全検索する必要はない。

【 0 0 4 9 】 以下、第1の実施の形態を図面を参照して詳細に説明する。但し、本発明の本質と関係の無いエラー処理等については説明を省略する。先ず、第1の実施の形態に係るコンピュータシステムの概略構成、並びに

50

コンピュータシステムが備えるファイルシステムの全体構成について説明する。

【0050】図2は第1の実施の形態に係るコンピュータシステムの概略構成を示すブロック図である。コンピュータシステムは、CPU110と、メモリ120と、外部記憶装置インターフェース130と、ファイルシステム140と、出力インターフェース150と、出力部160と、入力インターフェース170と、入力部180と、CPUバス190とを備えている。本例は、CPU110に対してオペレーティングシステム(OS)等を供給することによって、パーソナルコンピュータ及びワークステーション等のコンピュータを使用する例を示すものである。

【0051】上記各部の構成を詳述すると、CPU110は、CPUバス190を介してメモリ120～入力部180をアクセスし制御を行う。メモリ120は、CPUバス190を介してCPU110からアクセス可能な読み書き可能メモリ(RAM)であり、CPU110のワークメモリとして機能する。第1の実施の形態では、例えばハードディスク等の補助記憶装置(外部記憶装置)であるファイルシステム140等に格納され供給されるOSと該OS上で動く幾つかの応用プログラム141がメモリ120上に展開された状態で利用されるものとする。また、OSや応用プログラム141が使用するワーク領域121もメモリ120上に割り当てられる。

【0052】外部記憶装置インターフェース130は、ハードディスク等の外部記憶装置であるファイルシステム140に対するデータの入出力を扱う。ファイルシステム140は、上述したごとくオペレーティングシステム(OS)・応用プログラム141を格納する。出力インターフェース150は、出力部160に対してデータの表示・出力を実行。出力部160は、例えばCRTディスプレイ、LCD(液晶ディスプレイ)等の表示媒体、更にはプリンタ等の出力装置として構成されている。入力インターフェース1170は、入力部180を介してなされる利用者からの人力を受け取る。入力部180は、例えばキーボード、マウス等の入力装置として構成されている。

【0053】図1は第1の実施の形態に係るファイルシステムの全体構成を示すブロック図である。第1の実施の形態に係るファイルシステムは、カーネル/システムコール201(カーネル: OSの機能中で最も基本的な部分)と、圧縮要求発生手段203と、圧縮対象ファイル抽出手段204と、書き出し手段205と、ファイルオープン手段206と、ファイルクローズ手段207と、ファイル作成手段208と、ファイル削除手段209と、読み出し手段210と、圧縮手段211と、展開手段212と、インデックス並べ替え手段213と、inodeアクセス手段214と、書き出し手段215と、読み出し手段216と、ディスク装置217とを備

えている。

【0054】上記各部の構成を詳述すると、カーネル/システムコール201は、ファイルへのアクセス等を行う。圧縮要求発生手段203は、システムの時刻を計測する時間計測手段203aと、システムの負荷状況を調査する負荷計測手段203bとを有しており、一定時間毎にシステムの負荷状況を計測し負荷が軽かった場合か、ユーザ202からの強制実行要求があった場合に、圧縮処理要求を発生する。圧縮対象ファイル抽出手段204は、圧縮対象となるファイルを選び出す。書き出し手段205は、データの書き出し要求を処理する。

【0055】ファイルオープン手段206は、ファイルのオープン要求を処理する。ファイルクローズ手段207は、ファイルのクローズ要求を処理する。ファイル作成手段208は、ファイルの作成要求を処理する。ファイル削除手段209は、ファイルの削除要求を処理する。読み出し手段210は、データの読み出し要求を処理する。圧縮手段211は、ファイルのデータを圧縮する。展開手段212は、ファイルのデータを展開する。

【0056】インデックス並べ替え手段213は、inodeのリンク順序を繋ぎ替える。inodeアクセス手段214は、ディスク装置217上のinode領域217aを管理する。書き出し手段215は、ディスク装置217のData領域217bにデータを書き出す。読み出し手段216は、ディスク装置217のData領域217bからデータを読み出す。ディスク装置217は、inodeを保存するinode領域217aと、データを保存するData領域217bとを有する。

【0057】図3は第1の実施の形態に係るファイルシステム上のファイルが取り得る状態を表した状態遷移図である。通常状態302は、ファイルが圧縮も使用もされていない状態で、この状態からファイルオープン操作305により、ファイルがオープン(Read、Write、Appendなど全てのモード)された時に、使用中301の状態へ遷移する。使用中301の状態のファイルをファイルクローズ操作306で使用を終了すると、そのファイルは通常状態302に遷移する。

【0058】通常状態302から圧縮処理309を行うことにより、ファイルの状態は圧縮状態303に遷移する。圧縮状態303から展開処理+ファイルオープン操作307を行うと、使用中301に遷移する。また、通常状態302のファイルを削除処理310で削除すると、未使用状態304に遷移する。同様に、圧縮状態303のファイルを削除処理311で削除すると、未使用状態304に遷移する。ファイルの新規作成は、新規作成処理308によって行われ、未使用状態304のファイル(実体は存在しない)が通常状態302へと遷移する。

【0059】図4は第1の実施の形態に係るファイルシ

システムにおけるディスク装置217のi_node領域217aを模擬的に表すと共に、使用中301、通常状態302、圧縮状態303、未使用状態304の各状態に由来する使用中リスト416、通常状態リスト417、圧縮状態リスト419、未使用状態リスト420の関係の一例を表した図である。

【0060】図中i_node領域418には、番号付けされたi_node400、401、402、…がシーケンシャルに並んでいる。各i_nodeは、その順番とは独立して双方向リンクリストを構成するためのエントリを持つ。図示例では、現在圧縮状態のファイルのi_nodeは、圧縮状態リスト419に接続されたノード6(406)、ノード12(412)、ノード4(404)、及びノード14(414)である。

【0061】次に、上記のごとく構成してなる第1の実施の形態に係るファイルシステムにおけるファイルオープン処理、読み出し処理、書き出し処理、ファイルクローズ処理、ファイル作成処理、ファイル削除処理、圧縮要求発生処理、インデックス並べ替え処理、圧縮処理、展開処理、圧縮対象ファイル抽出処理について、図5乃至図19を参照して詳細に説明する。

【0062】図5及び図6は第1の実施の形態に係るカーネル/システムコール201からファイルオープン要求があった場合のファイルオープン手段206の処理の一例を示す流れ図である。先ず、i_nodeアクセス手段214よりi_node情報を読み込む(ステップS501)。次に、読み込んだi_node情報から当該ファイルの状態を調査する(ステップS502)。次に、そのファイルが圧縮状態である場合は(ステップS503の答がYES)、展開手段212で展開処理を行う(ステップS504)。ファイルが圧縮状態でない場合(ステップS503の答がNO)、展開が成功した場合は(ステップS505の答がYES)、ファイルのオープン作業とファイルオープン数のインクリメントを行う(ステップS506)。

【0063】次に、ファイルの状態属性を使用中に変更し(ステップS507)、インデックス並べ替え手段213を用いて当該ファイルのi_nodeを使用中リストの先頭に並べ替える(ステップS508)。但し、当該ファイルが既に使用中の場合も含む。即ち、同じファイルを同時に複数オープンすることが可能であり、それぞれファイル記述子で管理される。並べ替えが成功した場合は(ステップS509の答がYES)、本処理を終了する。並べ替えが失敗した場合(ステップS509の答がNO)、展開が失敗した場合は(ステップS505の答がNO)、エラー処理を行い(ステップS510)、本処理を終了する。

【0064】図7は第1の実施の形態に係るカーネル/システムコール201からデータの読み出し要求があつた場合の読み出し手段210の処理の一例を示す流れ図

である。先ず、i_nodeアクセス手段214よりi_node情報を読み込む(ステップS601)。次に、読み込んだi_node情報から当該ファイルの状態を調査する(ステップS602)。次に、そのファイルが(読み出し)オープンされている場合は(ステップS603の答がYES)、読み出し手段216によりデータを読み出す(ステップS604)。

【0065】読み出しが成功した場合は(ステップS605の答がYES)、本処理を終了する。ファイルが(読み出し)オープンされていない場合(ステップS603の答がNO)、読み出しが失敗した場合は(ステップS605の答がNO)、エラー処理を行い(ステップS606)、本処理を終了する。

【0066】図8は第1の実施の形態に係るカーネル/システムコール201からデータの書き出し要求があつた場合の書き出し手段205の処理の一例を示す流れ図である。先ず、i_nodeアクセス手段214よりi_node情報を読み込む(ステップS701)。次に、読み込んだi_node情報から当該ファイルの状態を調査する(ステップS702)。次に、そのファイルが(書き込みまたは追加モードで)オープンされている場合は(ステップS703の答がYES)、書き出し手段215によりデータを書き出す(ステップS704)。

【0067】書き出しが成功した場合は(ステップS705の答がYES)、本処理を終了する。書き出しが失敗した場合(ステップS705の答がNO)、ファイルが(書き込みまたは追加モードで)オープンされていない場合は(ステップS703の答がNO)、エラー処理を行い(ステップS706)、本処理を終了する。

【0068】図9及び図10は第1の実施の形態に係るカーネル/システムコール201からファイルクローズ要求があつた場合のファイルクローズ手段207の処理の一例を示す流れ図である。先ず、i_nodeアクセス手段214よりi_node情報を読み込む(ステップS801)。次に、読み込んだi_node情報から当該ファイルの状態を調査する(ステップS802)。次に、そのファイルがオープンされている場合は(ステップS803の答がYES)、当該ファイル記述子のファイルをクローズし、ファイルオープン数をデクリメントする(ステップS804)。

【0069】次に、その当該ファイルのファイルオープン数が0か、即ち、全てクローズされているかを調べ(ステップS805)、もし全てクローズされている場合は(ステップS805の答がYES)、ファイルの状態属性を通常状態に変更し(ステップS806)、インデックス並べ替え手段213を用いて当該ファイルのi_nodeを通常状態リストの先頭に並べ替える(ステップS807)。

【0070】並べ替えが成功した場合は(ステップS808の答がYES)、本処理を終了する。ファイルのオ

13

オープン数が0でない場合は(ステップS805の答がNO)、状態の変更は行わず本処理を終了する。並べ替えが失敗した場合(ステップS808の答がNO)、ファイルがオーブンされていない場合は(ステップS803の答がNO)、エラー処理を行い(ステップS809)、本処理を終了する。

【0071】図11は第1の実施の形態に係るカーネル／システムコール201からファイルの作成要求があつた場合のファイル作成手段208の処理の一例を示す流れ図である。先ず、i_nodeアクセス手段214よりi_node情報を読み込む(ステップS901)。次に、読み込んだi_node情報から新規ファイルが作成可能であるかを調査する(ステップS902)。新規ファイルが作成可能な場合は(ステップS903の答がYES)、新しいi_nodeとデータ領域を確保し(ステップS904)、ファイルの状態属性を通常状態にセットする(ステップS905)。

【0072】そして、インデックス並べ替え手段213により当該ファイルのi_nodeを通常状態リストの先頭に並べ替える(ステップS906)。並べ替えが成功した場合は(ステップS907の答がYES)、本処理を終了する。並べ替えが失敗した場合(ステップS907の答がNO)、新規ファイルが作成可能でない場合は(ステップS903の答がNO)、エラー処理を行い(ステップS908)、本処理を終了する。

【0073】図12は第1の実施の形態に係るカーネル／システムコール201からファイルの削除要求があつた場合のファイル削除手段209の処理の一例を示す流れ図である。先ず、i_nodeアクセス手段214よりi_node情報を読み込む(ステップS1000)。次に、読み込んだi_node情報からファイルの状態を調査する(ステップS1002)。ファイルの状態が使用中である場合は(ステップS1003の答がYES)、削除不可能なのでエラー処理を行い(ステップS1008)、本処理を終了する。

【0074】ファイルの状態が使用中でない場合は(ステップS1003の答がNO)、当該ファイルが使っていいたファイルシステム上のデータ領域を解放し(ステップS1004)、ファイルの状態属性を未使用に変更する(ステップS1005)。そして、インデックス並べ替え手段213により、当該ファイルのi_nodeを未使用状態リストの先頭に並べ替える(ステップS1006)。並べ替えが成功した場合は(ステップS1007の答がYES)、本処理を終了する。並べ替えが失敗した場合は(ステップS1007の答がNO)、エラー処理を行い(ステップS1008)、本処理を終了する。

【0075】図13及び図14は第1の実施の形態に係るファイル圧縮操作のきっかけとなる圧縮要求を発生する圧縮要求発生手段203の処理の一例を示す流れ図で

14

ある。先ず、待ち時間の初期値(システムで定められたデフォルト値があり、ユーザ202によっても変更可能)をセットする(ステップS1101)。そして、以下ステップS1102～ステップS1108を繰り返す。

【0076】先ず、次の3つの条件のうち1つ以上が成立した場合、次の処理に移行する(ステップS1102)。3つの条件とは、(1) 予めセットされている待ち時間を経過した場合、(2) ユーザ202からの強制実行要求があつた場合、(3) その他の要因(ディスクの空き領域が少なくなってきた等)で強制実行が必要になつた場合、である。但し、(1)の待ち時間は、時間計測手段203aによって計測される。(2)のユーザ202による強制実行要求は、ユーザ202が起動するコマンド等で通知されるものである。(3)はその他の要因であるが、本実施の形態では具体的に言及しない。

【0077】上記条件が満たされた場合、その要因が(1)の待ち時間が経過した場合かそれ以外の場合かを判断する(ステップS1103)。要因が(1)の場合には、負荷計測手段203bによってシステムの負荷状況を計測し(ステップS1104)、それが既定値(システムで定められたデフォルト値があり、ユーザ202によっても変更可能)よりも高いかを判断する(ステップS1105)。負荷が既定値より高い場合は(ステップS1105の答がYES)、実行を延期し、次のリトライ時間(システムで定められたデフォルト値があり、ユーザ202によっても変更可能)をセットし(ステップS1108)、ステップS1102に戻る。

【0078】負荷が既定値より低い場合(ステップS1105の答がNO)、要因が(2)または(3)の強制実行要求だった場合は(ステップS1103の答がYES)、圧縮対象ファイル抽出手段204による処理を実行する(ステップS1106)。そして、次の待ち時間(システムで定められたデフォルト値があり、ユーザ202によっても変更可能)をセットし(ステップS1107)、ステップS1102に戻る。

【0079】図15は第1の実施の形態に係るi_nodeリンクリストを並べ替える処理を行うインデックス並べ替え手段(i_nodeリスト並べ替え手段)213の処理の一例を示す流れ図である。先ず、並べ替えの対象となるファイルのi_nodeを変更前の状態のリストから抜き出し(ステップS1201)、抜き出したi_nodeを変更後の状態リストの先頭に挿入する(ステップS1202)。

【0080】次に、i_nodeアクセス手段214を通して、変更したi_node情報をディスク装置217のi_node領域217aへ書き出す(ステップS1203)。書き出しが成功した場合は(ステップS1204の答がYES)、本処理を終了する。書き出しが失敗した場合は(ステップS1204の答がNO)、エ

15

ラー処理を行い(ステップS1205)、本処理を終了する。

【0081】図16は第1の実施の形態に係る圧縮対象ファイル抽出手段204からの圧縮要求があった場合の圧縮手段211の処理の一例を示す流れ図である。先ず、指定されたファイルのデータを読み出し手段216で読み出し(ステップS1301)、読み出しが成功した場合は(ステップS1302の答がYES)、読み出したデータを圧縮プログラムで圧縮し(ステップS1303)、書き出し手段215でディスク装置217のデータ領域217bにデータを書き出す(ステップS1304)。

【0082】書き出しが成功した場合は(ステップS1305の答がYES)、本処理を終了する。書き出しが失敗した場合(ステップS1305の答がNO)、読み出しが失敗した場合は(ステップS1302の答がNO)、エラー処理を行い(ステップS1306)、本処理を終了する。

【0083】図17は第1の実施の形態に係るファイルオープン手段206からの展開要求があった場合の展開手段212の処理の一例を示す流れ図である。先ず、指定されたファイルのデータを読み出し手段216で読み出し(ステップS1401)、読み出しが成功した場合は(ステップS1402の答がYES)、読み出したデータを展開プログラムで展開し(ステップS1403)、書き出し手段215でディスク装置217のデータ領域217bにデータを書き出す(ステップS1404)。

【0084】書き出しが成功した場合は(ステップS1405の答がYES)、本処理を終了する。書き出しが失敗した場合(ステップS1405の答がNO)、読み出しが失敗した場合は(ステップS1402の答がNO)、エラー処理を行い(ステップS1406)、本処理を終了する。

【0085】図18及び図19は第1の実施の形態に係る圧縮要求発生手段203からの圧縮要求があった場合の圧縮対象ファイル抽出手段204の処理の一例を示す流れ図である。先ず、i_nodeアクセス手段214よりi_nodeの通常状態リストを読み込む(ステップS1501)。以降、ステップS1502～ステップS1507を繰り返す。このループ内で、読み込んだ通常状態リストを末尾から順次選び出していく(ステップS1502)。リストの先頭に達した場合は(ステップS1503の答がYES)、ループを抜け本処理を終了する。

【0086】また、選び出されたi_nodeのファイルの最終アクセス時刻とシステムの現在時刻との差が既定値(システムで定められたデフォルト値があり、ユーザ202によっても変更可能)以上である場合は(ステップS1504の答がYES)、そのファイルが圧縮可

10

能ファイルか否かを調べる(ステップS1505)。デバイススペシャルファイル等の圧縮不可能なファイルの場合は、処理をせずにステップS1502に戻る。

【0087】圧縮可能なファイルの場合は、圧縮手段211を呼び出し、実際の圧縮処理を行い(ステップS1506)、ファイルの状態属性を圧縮状態に変更し(ステップS1507)、インデックス並べ替え手段213を呼び出す(ステップS1508)。並べ替えが成功した場合は(ステップS1509の答がYES)、ステップS1502に戻る。並べ替えが失敗した場合は(ステップS1509の答がNO)、エラー処理を行い(ステップS1510)、本処理を終了する。ファイルの最終アクセス時刻とシステムの現在時刻との差が既定値以下になった場合は(ステップS1504の答がNO)、それよりも前にリンクされている全てのノードの時刻の差は既定値以下となるため、その時点でループを抜けて本処理を終了する。

20

【0088】上述したように、第1の実施の形態によれば、ファイルシステムは、ファイルのアクセス発生時に当該ファイルのインデックスの順番を先頭に並べ替えるインデックス並べ替え手段213と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する圧縮対象ファイル抽出手段204と、システムの負荷状況を考慮して圧縮処理を実行するか延期するかを決定する圧縮要求発生手段203と、ファイルを条件によって圧縮する圧縮手段211と、アクセス要求発生時にファイルの状態を元に戻す展開手段212とを有すると共に、ファイルのインデックスを双方向リンクリストを用いて実現しているため、ディスクの使用効率を向上できると共に、圧縮・展開にかかるオーバヘッドを抑制することができ、また、アプリケーションの使用感を損なうことが少ない。

30

【0089】即ち、ファイルシステムは、個々のファイルは実体(データ本体)と、その実体の位置を指示するインデックスから構成され、それぞれを別々の領域で管理する手段を持つ。つまり、個々のファイルのインデックスの集合(インデックス列)はインデックス領域に格納され、実体はデータ領域に格納される。また、或るファイルに対するアクセス(Read, Write, Append等)が発生した時に、そのファイルのインデックスの順番を先頭に並べ替えるインデックス並べ替え手段213を備えており、常に最終アクセス時間が早い順になっている。

40

【0090】加えて、最終アクセス順に並べられたインデックスから最終アクセス時間が古い順に取り出す圧縮対象ファイル抽出手段204を備え、その取り出されたファイルを条件(最終アクセス時間、ファイルの種類、その他ファイル固有の属性)によって圧縮する圧縮手段211と、圧縮されたファイルに対してアクセス要求が生じた時に、状態を元に戻す(展開する)展開手段21

50

17

2とを備える。

【 0 0 9 1 】 また、圧縮要求発生手段2 0 3 は、システムの時刻を計測する時間計測手段2 0 3 a とシステムの負荷状態を調査する負荷計測手段2 0 3 b を用いて実現される。圧縮要求発生手段2 0 3 は、圧縮対象ファイル抽出手段2 0 4 に対して処理要求を発生する。処理要求はユーザから直接発行される場合もある。圧縮要求発生手段2 0 3 は、そのシステムの負荷状況を考慮して圧縮処理を実行するか延期するかを決定する。即ち、負荷が高い場合は要求を無視するかスケジューリングをやり直す。

【 0 0 9 2 】 このように、ファイルのアクセス頻度に応じて個々のファイルを圧縮することにより、ディスクの使用効率を向上させると共に、頻繁にアクセスされるファイルに関しては圧縮しないという方法をとることにより、圧縮・展開にかかる時間的、CPU負荷的なオーバヘッドを抑制することが可能となる効果を奏する。また、ファイルの圧縮処理をシステムのアイドル時間(負荷が低い時)に行うため、アプリケーションを使用中のユーザにとってその使用感を損なうことが少ないという効果を奏する。

【 0 0 9 3 】 [2] 第2の実施の形態

本発明の第2の実施の形態では、上記第1の実施の形態と同様に、例えばUNIXのファイルシステムに適用した例を示す。上記第1の実施の形態ではファイルのインデックス列を双向リンクリストを用いて実現しているが、第2の実施の形態ではファイルのインデックス列を円形配列を用いて同様なシステムを実現した例を挙げる。

【 0 0 9 4 】 第2の実施の形態に係るファイルシステムは、カーネル/システムコール2 0 1 と、圧縮要求発生手段2 0 3 と、圧縮対象ファイル抽出手段2 0 4 と、書き出し手段2 0 5 と、ファイルオープン手段2 0 6 と、ファイルクローズ手段2 0 7 と、ファイル作成手段2 0 8 と、ファイル削除手段2 0 9 と、読み出し手段2 1 0 と、圧縮手段2 1 1 と、展開手段2 1 2 と、インデックス並べ替え手段2 1 3 と、inodeアクセス手段2 1 4 と、書き出し手段2 1 5 と、読み出し手段2 1 6 と、ディスク装置2 1 7 とを備えている(上記図1参照)。

【 0 0 9 5 】 第2の実施の形態に係るファイルシステムの基本的な構成は上記第1の実施の形態と同じであるが、上記図1におけるインデックス並べ替え手段2 1 3 、圧縮対象ファイル抽出手段2 0 4 、及びディスク装置2 1 7 のinode領域2 1 7 a でのデータ格納方式が異なる。従って、上記図4のinode領域2 1 7 a の模擬図に対応する第2の実施の形態のデータ格納方式の模擬図と、インデックス並べ替え手段2 1 3 に対応するinode配列並べ替え手段の流れ図、及び第2の実施の形態に対応した圧縮対象ファイル抽出手段の

10

20

30

並べ替え手段(インデックス並べ替え手段2 1 3 に対応)の処理の一例を示す流れ図である。先ず、並べ替えの対象となるファイルのinodeを変更前の状態の円形配列から抜き出し(ステップS 1 7 0 1)、その配列の先頭ポインタの位置が空き要素でない場合は(ステップS 1 7 0 2 の答がNO)、上記図2 0 中のポインタ1 6 0 5 のように変更後の状態配列の先頭ポインタを1つ前に戻す(ステップS 1 7 0 3)。先頭ポインタの位置が空き要素の場合は(ステップS 1 7 0 2 の答がYES)、ステップS 1 7 0 4 へ移行する。

【 0 1 0 0 】 先頭ポインタと末尾ポインタが重なった場合は(ステップS 1 7 0 4 の答がYES)、円形配列中の空き要素をつめる作業を行う必要が生じるため、先頭ポインタから後方へ空き要素を検索していく、見つかった場合は、後方にある非空き要素をそこへ移動させる操作を、末尾ポインタの位置まで繰り返し行い、上記図2 0 中のポインタ1 6 0 6 のように最終的な末尾要素の位置へ末尾ポインタを再設定する。この操作により、円形配列の非空き要素は連続して並ぶ(ステップS 1 7 0 5)。

【 0 1 0 1 】 先頭ポインタと末尾ポインタが重ならなかった場合(ステップS 1 7 0 4 の答がNO)か、ステップS 1 7 0 5 が終了した場合は、先頭ポインタの位置にステップS 1 7 0 1 で取り出したinode番号を格

18

流れ図を追加し、詳細に説明する。

【 0 0 9 6 】 図2 0 は第2の実施の形態に係るinode領域2 1 7 a を模擬的に表した図である。上記第1の実施の形態で実現した双向リンクリストに対応するものであり、リング状の配列(円形配列1 6 0 2)を用いて実装した例である。図中inode領域1 6 0 1 には番号付けされたinodeがシーケンシャルに並んでいる。円形配列1 6 0 2 は、このinode領域1 6 0 1 に存在するinodeの数と同数の要素数を持つものとする。

【 0 0 9 7 】 そして、配列の開始点を指示する先頭ポインタ1 6 0 3 と、終了点を指示する末尾ポインタ1 6 0 4 を備える。この円形配列1 6 0 2 の要素には、図示のごとくinode番号を格納する。図の空欄は、空き要素でまだ使用されていない状態か、格納されていたinode番号が取り出された状態である。

【 0 0 9 8 】 図中には1つの円形配列しか示していないが、実際には使用中3 0 1 、通常状態3 0 2 、圧縮状態3 0 3 、未使用状態3 0 4 の各状態に由来する使用中配列、通常状態配列、圧縮状態配列、未使用状態配列の4つの円形配列が存在する。図示例では、この円形配列1 6 0 2 が表現するinodeの順番は、1 0 、2 5 、5 、1 3 、2 4 、2 、0 、… 2 1 、2 3 、1 2 、1 6 、6 である。

【 0 0 9 9 】 図2 1 及び図2 2 は第2の実施の形態に係る円形配列を並べ替える処理を行うinode配列並べ替え手段(インデックス並べ替え手段2 1 3 に対応)の処理の一例を示す流れ図である。先ず、並べ替えの対象となるファイルのinodeを変更前の状態の円形配列から抜き出し(ステップS 1 7 0 1)、その配列の先頭ポインタの位置が空き要素でない場合は(ステップS 1 7 0 2 の答がNO)、上記図2 0 中のポインタ1 6 0 5 のように変更後の状態配列の先頭ポインタを1つ前に戻す(ステップS 1 7 0 3)。先頭ポインタの位置が空き要素の場合は(ステップS 1 7 0 2 の答がYES)、ステップS 1 7 0 4 へ移行する。

【 0 1 0 0 】 先頭ポインタと末尾ポインタが重なった場合は(ステップS 1 7 0 4 の答がYES)、円形配列中の空き要素をつめる作業を行う必要が生じるため、先頭ポインタから後方へ空き要素を検索していく、見つかった場合は、後方にある非空き要素をそこへ移動させる操作を、末尾ポインタの位置まで繰り返し行い、上記図2 0 中のポインタ1 6 0 6 のように最終的な末尾要素の位置へ末尾ポインタを再設定する。この操作により、円形配列の非空き要素は連続して並ぶ(ステップS 1 7 0 5)。

【 0 1 0 1 】 先頭ポインタと末尾ポインタが重ならなかった場合(ステップS 1 7 0 4 の答がNO)か、ステップS 1 7 0 5 が終了した場合は、先頭ポインタの位置にステップS 1 7 0 1 で取り出したinode番号を格

50

19

納する(ステップS1706)。次に、inodeアクセス手段214を通して、変更したinode情報をディスク装置217のinode領域217aへ書き出す(ステップS1707)。書き出しが成功した場合は、本処理を終了する。書き出しが失敗した場合は、エラー処理を行い(ステップS1709)、本処理を処理する。

【0102】図23及び図24は第2の実施の形態に係る圧縮要求発生手段203からの圧縮要求があつた場合の圧縮対象ファイル抽出手段204の処理の一例を示す流れ図である。先ず、inodeアクセス手段214よりinodeの通常状態配列を読み込む(ステップS1801)。以降、ステップS1802～1810を繰り返す。このループ内で、読み込んだ通常状態配列を末尾ポインタから先頭へ順次選び出していく(ステップS1802)。

【0103】先頭ポインタに達した場合は(ステップS1803の答がYES)、ループを抜け本処理を終了する。また、選び出した配列の要素が空き要素だった場合は(ステップS1804の答がYES)、ステップS1802に戻る。次に、選び出されたinodeのファイルの最終アクセス時刻とシステムの現在時刻との差が既定値(システムで定められたデフォルト値があり、ユーザ202によっても変更可能)以上である場合は(ステップS1805の答がYES)、そのファイルが圧縮可能ファイルか否かを調べる(ステップS1806)。

【0104】デバイススペシャルファイル等の圧縮不可能なファイルの場合は、圧縮手段211を呼び出し、実際の圧縮処理を行い(ステップS1807)、ファイルの状態属性を圧縮状態にし(ステップS1808)、inode配列並べ替え手段213を呼び出す(ステップS1809)。並べ替えが成功した場合は(ステップS1810の答がYES)、ステップS1802に戻る。

【0105】並べ替えが失敗した場合は(ステップS1810の答がNO)、エラー処理を行い(ステップS1811)、本処理を終了する。ファイルの最終アクセス時刻とシステムの現在時刻との差が既定値以下になった場合は(ステップS1805の答がNO)、それよりも前に並んでいる全てのノードの時刻の差は既定値以下となるため、その時点でのループを抜けて本処理を終了する。

【0106】上述したように、第2の実施の形態によれば、ファイルシステムは、ファイルのアクセス発生時に当該ファイルのインデックスの順番を先頭に並べ替えるインデックス並べ替え手段213と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する圧縮対象ファイル抽出手段204と、システムの負荷状況を考慮して圧縮処理を実行するか延期するかを決定する圧縮要求発生手段203と、フ

20

イルを条件によって圧縮する圧縮手段211と、アクセス要求発生時にファイルの状態を元に戻す展開手段212とを有すると共に、ファイルのインデックスを凸形配列を用いて実現しているため、ディスクの使用効率を向上できると共に、圧縮・展開にかかるオーバヘッドを抑制でき、また、アプリケーションの使用感を損なうことが少ない。

【0107】即ち、ファイルのアクセス頻度に応じて個々のファイルを圧縮することにより、ディスクの使用効率を向上させると共に、頻繁にアクセスされるファイルに関しては圧縮しないという方法をとることにより、圧縮・展開にかかる時間的、CPU負荷的なオーバヘッドを抑制することが可能となる効果を奏する。また、ファイルの圧縮処理をシステムのアイドル時間(負荷が低い時)に行うため、アプリケーションを使用中のユーザにとってその使用感を損なうことが少ないという効果を奏する。

【0108】尚、本発明は、複数の機器から構成されるシステムに適用しても、1つの機器からなる装置に適用してもよい。前述した実施形態の機能を実現するソフトウェアのプログラムコードを記憶した記憶媒体を、システム或いは装置に供給し、そのシステム或いは装置のコンピュータ(またはCPUやMPU)が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

【0109】この場合、記憶媒体から読み出されたプログラムコード自身が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0110】プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0111】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOSなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0112】更に、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0113】

21

【発明の効果】以上説明したように、請求項1の発明によれば、ファイルを蓄積し管理するファイル管理装置であって、ファイルのアクセス発生時に当該ファイルの実体の位置を示すインデックスの順番を先頭に並べ替える並べ替え手段と、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する抽出手段と、抽出したファイルの状態を条件に応じて変更する変更手段とを有するため、ファイルのアクセス頻度に応じて個々のファイルを変更(圧縮)することにより、記憶媒体(ディスク)の使用効率を向上させることができる。また、頻繁にアクセスされるファイルに関しては圧縮しないという方法をとることにより、圧縮・展開にかかる時間的、C P U負荷的なオーバヘッドを抑制することができる。

【0114】請求項2の発明によれば、ファイルが利用されるシステムの負荷状況を計測する負荷計測手段と、時刻を計測する時間計測手段とを有し、前記変更手段は、ファイルの状態変更をシステムの負荷が軽い時間に行うため、ファイルの状態変更である圧縮処理をシステムのアイドル時間(負荷が低い時)に行うことにより、アプリケーションを使用中のユーザにとってその使用感を損なうことが少なくなる。

【0115】請求項3の発明によれば、前記ファイルの状態とは、ファイルがオープンされた使用中、ファイルが使用中でもなく圧縮もされていない通常状態、ファイルが圧縮された圧縮状態、ファイルが利用されていない未使用状態であるため、使用中、通常状態、圧縮状態、未使用状態の何れかの状態にあるファイルを条件によって変更することで、記憶媒体(ディスク)の使用効率の向上、圧縮・展開にかかる時間的、C P U負荷的なオーバヘッドの抑制を図ることができる。

【0116】請求項4の発明によれば、前記並べ替え手段は、通常状態のファイルにオープン要求があった場合はそのインデックスを使用中列の先頭に移動させ、使用中のファイルにクローズ要求があった場合はそのインデックスを通常状態列の先頭に移動させ、通常状態のファイルに圧縮要求があった場合はそのインデックスを圧縮状態列の先頭に移動させ、圧縮状態のファイルにアクセス要求があった場合はそのインデックスを使用中列の先頭に移動させ、通常状態或いは圧縮状態のファイルに削除要求があった場合はそのインデックスを未使用状態列の先頭に移動させるため、各状態にあるファイルへのオープン要求、クローズ要求、圧縮要求、アクセス要求、削除要求に高速に対処することができる。

【0117】請求項5の発明によれば、前記変更手段は、最終アクセス時刻、ファイルの種類等の条件に応じてファイルを圧縮するため、ファイルのアクセス頻度に応じて個々のファイルを圧縮することで、記憶媒体(ディスク)の使用効率を向上させることができる。

【0118】請求項6の発明によれば、各ファイルのイ

22

ンデックス列は双方向リンクリストを用いるため、双方向リンクリストを用いてインデックスを並べ替えることで、記憶媒体(ディスク)の使用効率の向上、オーバヘッドの抑制を図ることができる。

【0119】請求項7の発明によれば、各ファイルのインデックス列は円形配列を用いるため、円形配列を用いてインデックスを並べ替えることで、記憶媒体(ディスク)の使用効率の向上、オーバヘッドの抑制を図ることができる。

10 【0120】請求項8の発明によれば、ファイルを蓄積し管理するファイル管理方法であって、ファイルのアクセス発生時に当該ファイルの実体の位置を示すインデックスの順番を先頭に並べ替える並べ替えステップと、最終アクセス順に並べられたファイルのインデックスから最終アクセス時間が古い順に抽出する抽出ステップと、抽出したファイルの状態を条件に応じて変更する変更ステップとを有するため、請求項1の発明と同様に、記憶媒体(ディスク)の使用効率を向上させることができると共に、圧縮・展開にかかる時間的、C P U負荷的なオーバヘッドを抑制することができる。

20 【0121】請求項9の発明によれば、ファイルが利用されるシステムの負荷状況を計測する負荷計測ステップと、時刻を計測する時間計測ステップとを有し、前記変更ステップでは、ファイルの状態変更をシステムの負荷が軽い時間に行うため、請求項2の発明と同様に、ファイルの状態変更である圧縮処理をシステムのアイドル時間(負荷が低い時)に行うことにより、アプリケーションを使用中のユーザにとってその使用感を損なうことが少なくなる。

30 【0122】請求項10の発明によれば、前記ファイルの状態とは、ファイルがオープンされた使用中、ファイルが使用中でもなく圧縮もされていない通常状態、ファイルが圧縮された圧縮状態、ファイルが利用されていない未使用状態であるため、請求項3の発明と同様に、使用中、通常状態、圧縮状態、未使用状態の何れかの状態にあるファイルを条件によって変更することで、記憶媒体(ディスク)の使用効率の向上、圧縮・展開にかかる時間的、C P U負荷的なオーバヘッドの抑制を図ることができる。

40 【0123】請求項11の発明によれば、前記並べ替えステップでは、通常状態のファイルにオープン要求があった場合はそのインデックスを使用中列の先頭に移動させ、使用中のファイルにクローズ要求があった場合はそのインデックスを通常状態列の先頭に移動させ、通常状態のファイルに圧縮要求があった場合はそのインデックスを圧縮状態列の先頭に移動させ、圧縮状態のファイルにアクセス要求があった場合はそのインデックスを使用中列の先頭に移動させ、通常状態或いは圧縮状態のファイルに削除要求があった場合はそのインデックスを未使用状態列の先頭に移動させるため、請求項4の発明と同

50

様に、各状態にあるファイルへのオープン要求、クローズ要求、圧縮要求、アクセス要求、削除要求に高速に対処することができる。

【 0 1 2 4 】請求項1 2 の発明によれば、前記変更ステップでは、最終アクセス時刻、ファイルの種類等の条件に応じてファイルを圧縮するため、請求項5 の発明と同様に、ファイルのアクセス頻度に応じて個々のファイルを圧縮することで、記憶媒体(ディスク)の使用効率を向上させることができる。

【 0 1 2 5 】請求項1 3 の発明によれば、各ファイルのインデックス列は双向リンクリストを用いるため、請求項6 の発明と同様に、双向リンクリストを用いてインデックスを並べ替えることで、記憶媒体(ディスク)の使用効率の向上、オーバヘッドの抑制を図ることができる。

【 0 1 2 6 】請求項1 4 の発明によれば、各ファイルのインデックス列は円形配列を用いるため、請求項7 の発明と同様に、円形配列を用いてインデックスを並べ替えることで、記憶媒体(ディスク)の使用効率の向上、オーバヘッドの抑制を図ることができる。

【 図面の簡単な説明】

【 図1 】本発明の第1 及び第2 の実施の形態に係るファイルシステムの全体構成を示すブロック図である。

【 図2 】本発明の第1 及び第2 の実施の形態に係るファイルシステムを備えたコンピュータシステムの概略構成を示すブロック図である。

【 図3 】本発明の第1 の実施の形態に係るファイルシステム上のファイルが取り得る状態の遷移を示す説明図である。

【 図4 】本発明の第1 の実施の形態に係るファイルシステムにおけるディスク装置のi n o d e 領域を模擬的に表すと共に、使用中、通常状態、圧縮状態、未使用状態の各状態に由来する使用中リスト、通常状態リスト、圧縮状態リスト、未使用状態リストの関係の一例を示す説明図である。

【 図5 】本発明の第1 の実施の形態に係るカーネル／システムコールからファイルオープン要求があった場合のファイルオープン手段の処理の一例を示す流れ図である。

【 図6 】本発明の第1 の実施の形態に係るカーネル／システムコールからファイルオープン要求があった場合のファイルオープン手段の処理の一例を示す流れ図である。

【 図7 】本発明の第1 の実施の形態に係るカーネル／システムコールからデータの読み出し要求があった場合の読み出し手段の処理の一例を示す流れ図である。

【 図8 】本発明の第1 の実施の形態に係るカーネル／システムコールからデータの書き出し要求があった場合の書き出し手段の処理の一例を示す流れ図である。

【 図9 】本発明の第1 の実施の形態に係るカーネル／シ

ステムコールからファイルクローズ要求があった場合のファイルクローズ手段の処理の一例を示す流れ図である。

【 図1 0 】本発明の第1 の実施の形態に係るカーネル／システムコールからファイルクローズ要求があった場合のファイルクローズ手段の処理の一例を示す流れ図である。

【 図1 1 】本発明の第1 の実施の形態に係るカーネル／システムコールからファイルの作成要求があった場合のファイル作成手段の処理の一例を示す流れ図である。

【 図1 2 】本発明の第1 の実施の形態に係るカーネル／システムコールからファイルの削除要求があった場合のファイル削除手段の処理の一例を示す流れ図である。

【 図1 3 】本発明の第1 の実施の形態に係るファイル圧縮操作のきっかけとなる圧縮要求を発生する圧縮要求発生手段の処理の一例を示す流れ図である。

【 図1 4 】本発明の第1 の実施の形態に係るファイル圧縮操作のきっかけとなる圧縮要求を発生する圧縮要求発生手段の処理の一例を示す流れ図である。

20 【 図1 5 】本発明の第1 の実施の形態に係るi n o d e リンクリストを並べ替える処理を行うインデックス並べ替え手段の処理の一例を示す流れ図である。

【 図1 6 】本発明の第1 の実施の形態に係る圧縮対象ファイル抽出手段からの圧縮要求があった場合の圧縮手段の処理の一例を示す流れ図である。

【 図1 7 】本発明の第1 の実施の形態に係るファイルオープン手段からの展開要求があった場合の展開手段の処理の一例を示す流れ図である。

30 【 図1 8 】本発明の第1 の実施の形態に係る圧縮要求発生手段からの圧縮要求があった場合の圧縮対象ファイル抽出手段の処理の一例を示す流れ図である。

【 図1 9 】本発明の第1 の実施の形態に係る圧縮要求発生手段からの圧縮要求があった場合の圧縮対象ファイル抽出手段の処理の一例を示す流れ図である。

【 図2 0 】本発明の第2 の実施の形態に係るi n o d e 領域を模擬的に表した説明図である。

【 図2 1 】本発明の第2 の実施の形態に係る円形配列を並べ替える処理を行うインデックス並べ替え手段の処理の一例を示す流れ図である。

【 図2 2 】本発明の第2 の実施の形態に係る円形配列を並べ替える処理を行うインデックス並べ替え手段の処理の一例を示す流れ図である。

【 図2 3 】本発明の第2 の実施の形態に係る圧縮要求発生手段からの圧縮要求があった場合の圧縮対象ファイル抽出手段の処理の一例を示す流れ図である。

【 図2 4 】本発明の第2 の実施の形態に係る圧縮要求発生手段からの圧縮要求があった場合の圧縮対象ファイル抽出手段の処理の一例を示す流れ図である。

【 符号の説明】

25

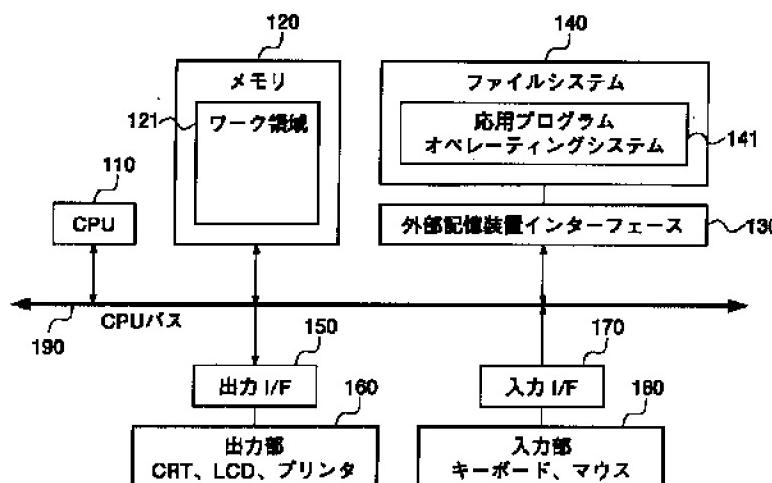
- 203 圧縮要求発生手段
 203a 時間計測手段
 203b 負荷計測手段
 204 圧縮対象ファイル抽出手段
 205 書き出し手段
 206 ファイルオープン手段
 207 ファイルクローズ手段
 208 ファイル作成手段
 209 ファイル削除手段
 210 読み出し手段

- 211 圧縮手段
 212 展開手段
 213 インデックス並べ替え手段
 214 i_node アクセス手段
 215 書き出し手段
 216 読み出し手段
 217 ディスク装置
 217a i_node 領域
 217b Data 領域

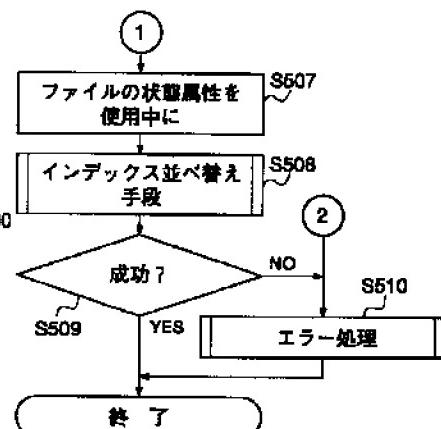
10

26

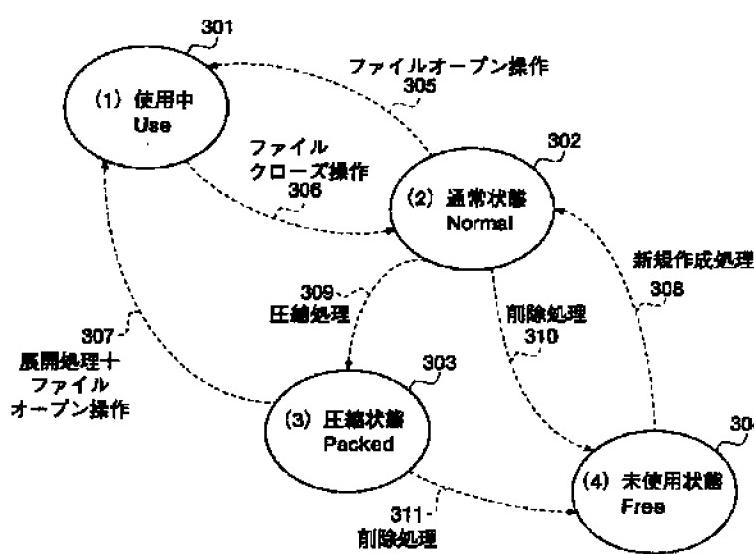
【 図2 】



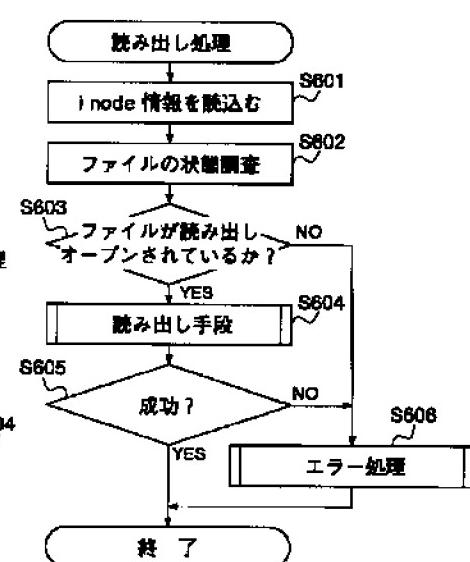
【 図6 】



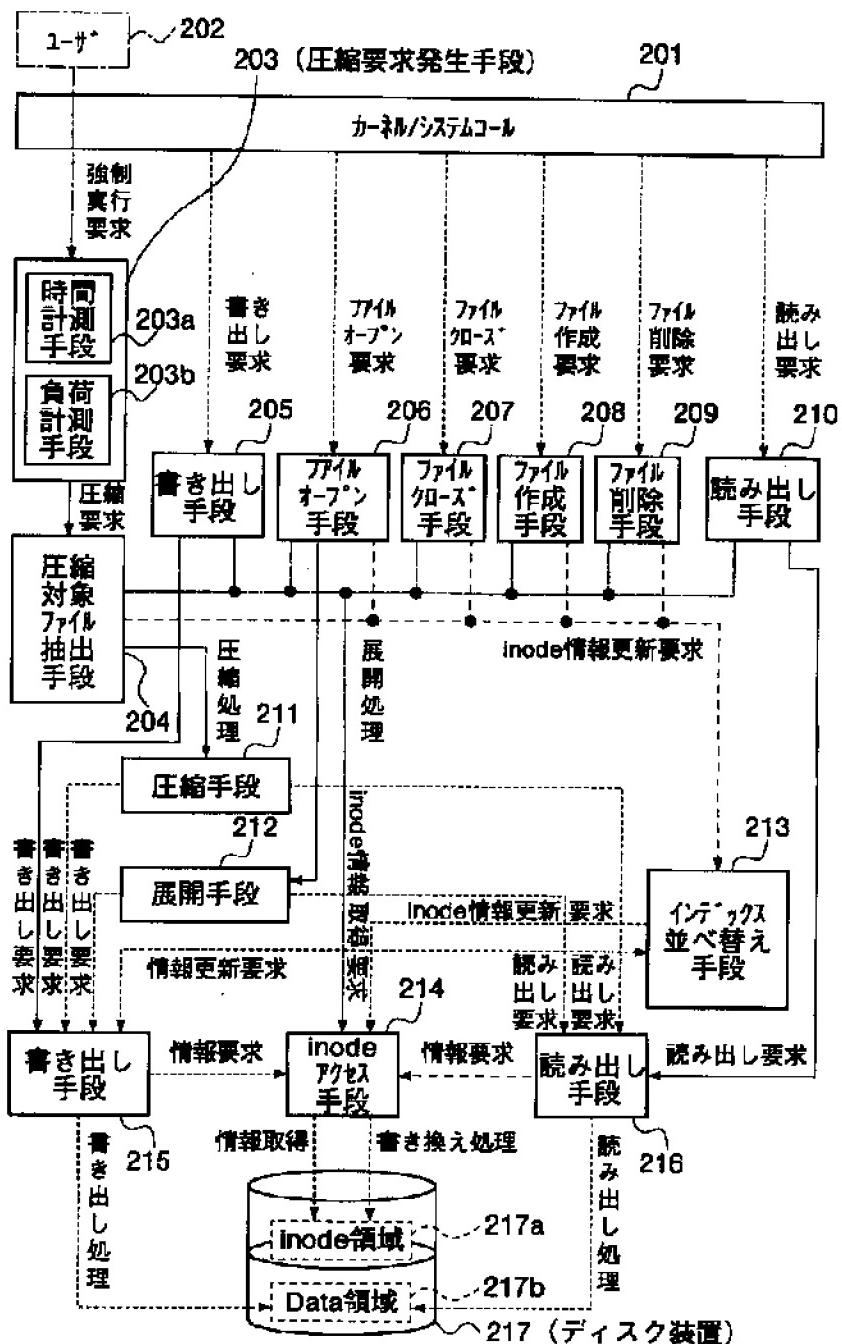
【 図3 】



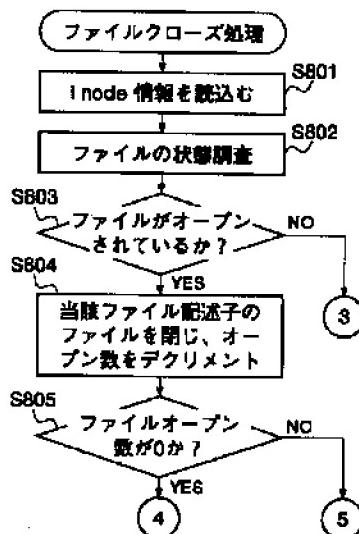
【 図7 】



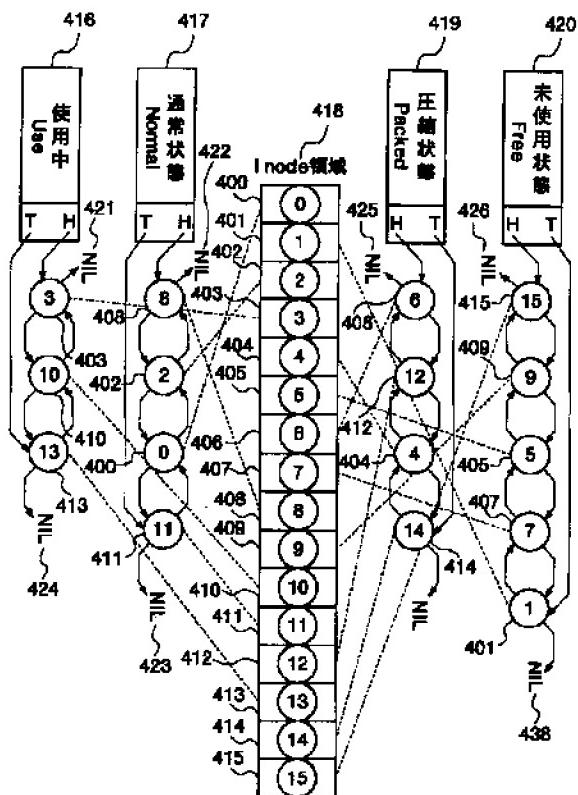
[図 1]



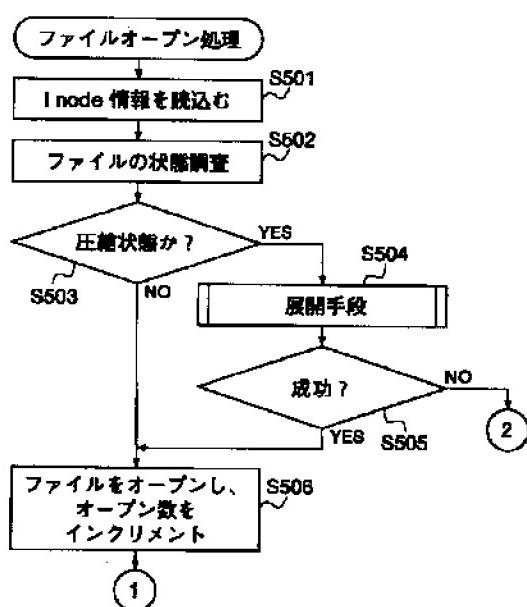
[圖9]



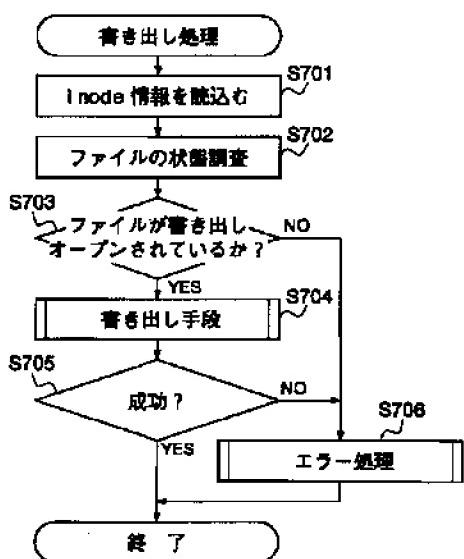
【 図4 】



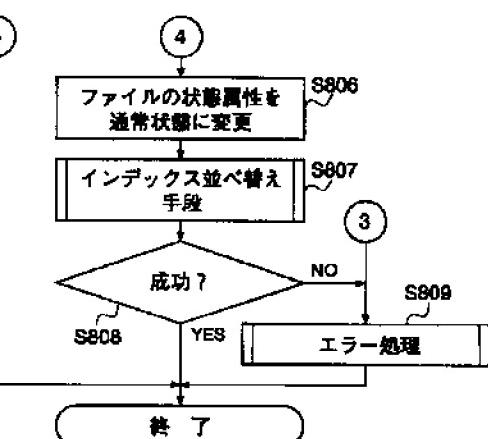
【 図5 】



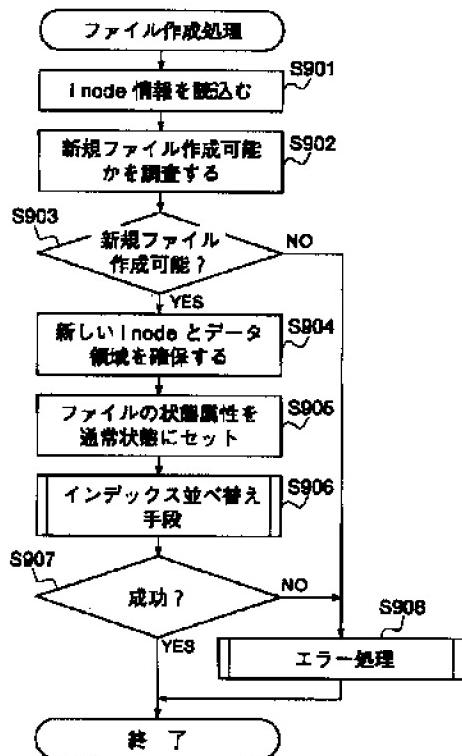
【 図8 】



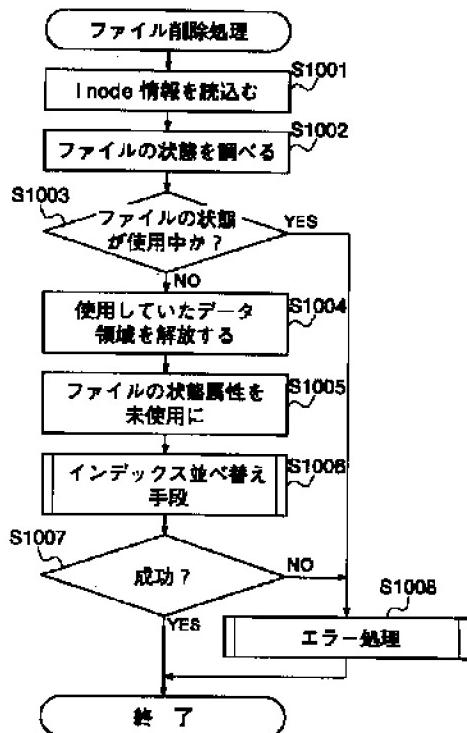
【 図10 】



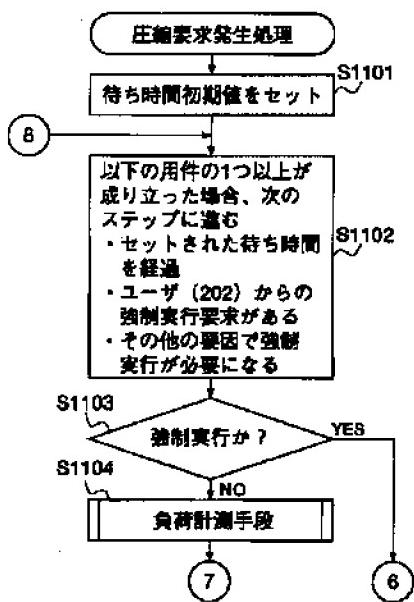
【 図1 1 】



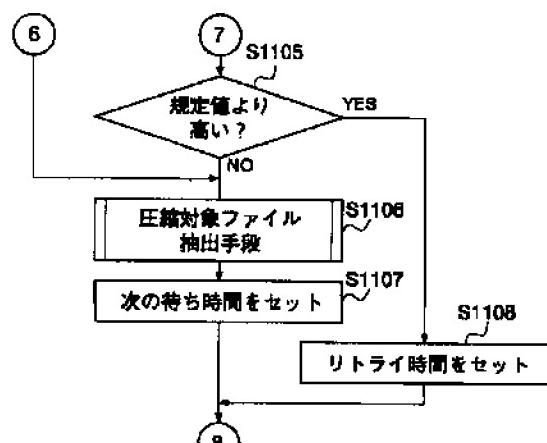
【 図1 2 】



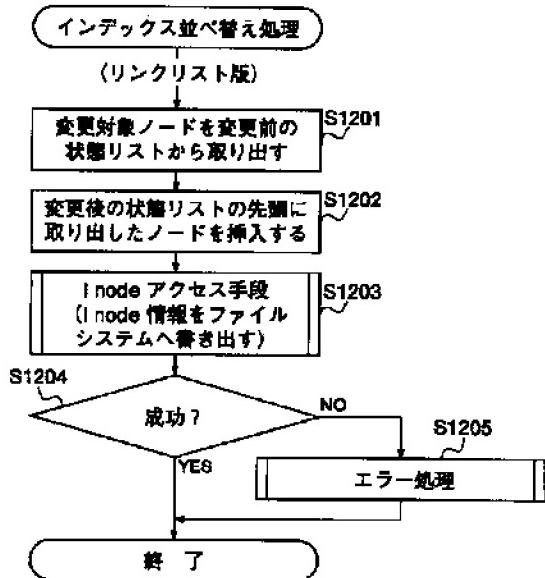
【 図1 3 】



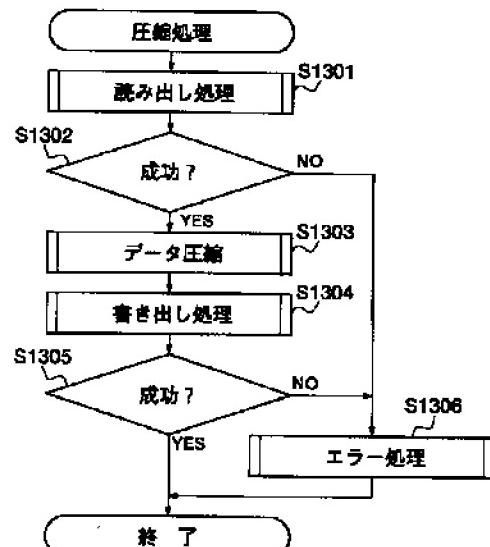
【 図1 4 】



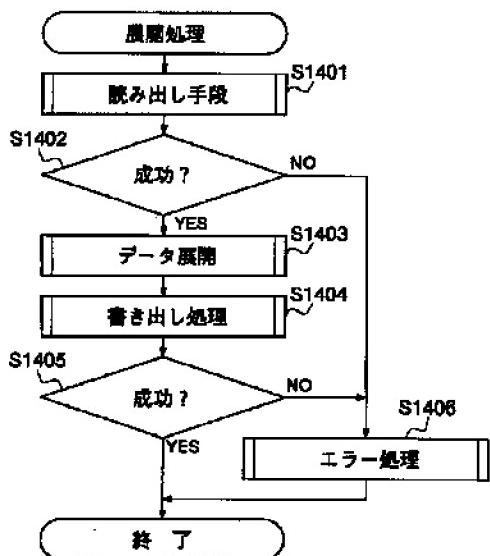
【 図1 5 】



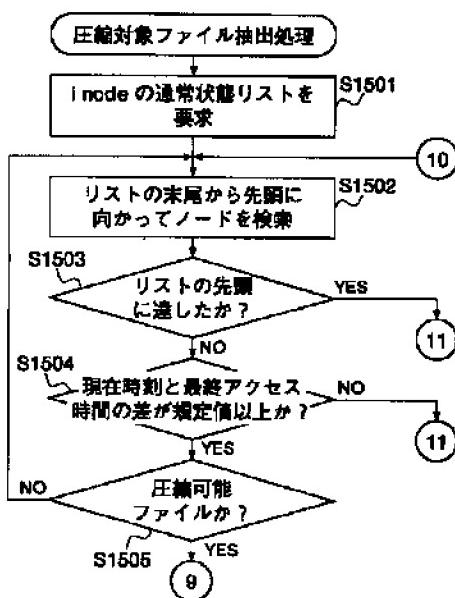
【 図1 6 】



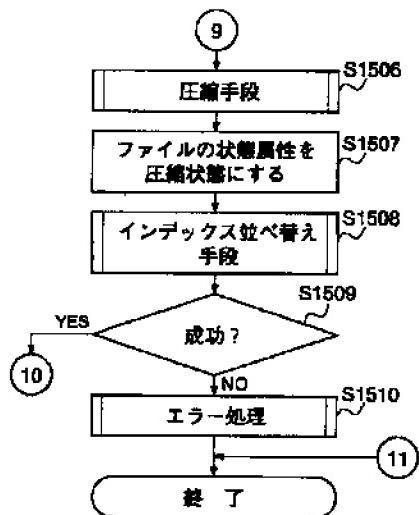
【 図1 7 】



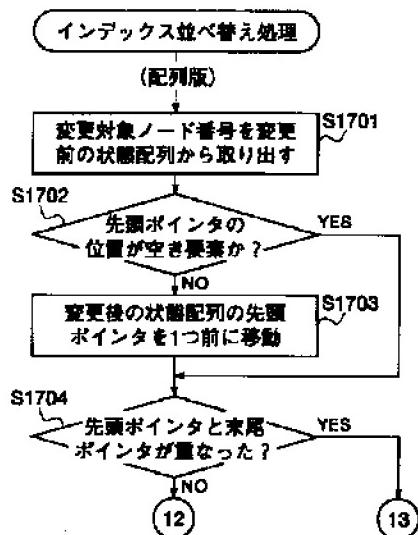
【 図1 8 】



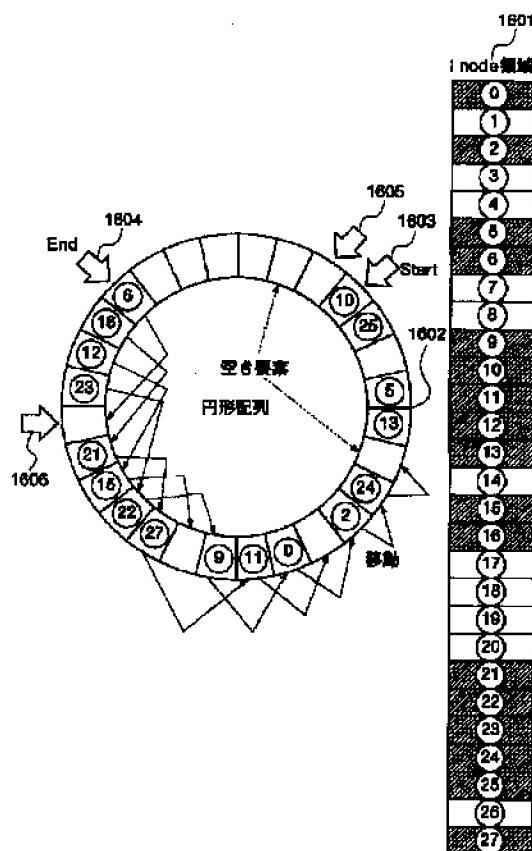
【 図19 】



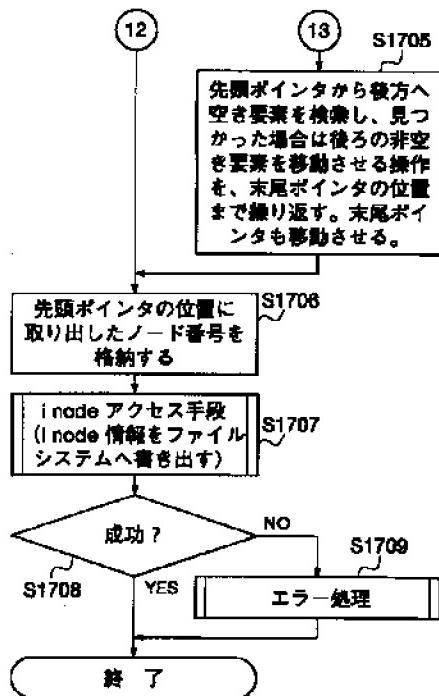
【 図21 】



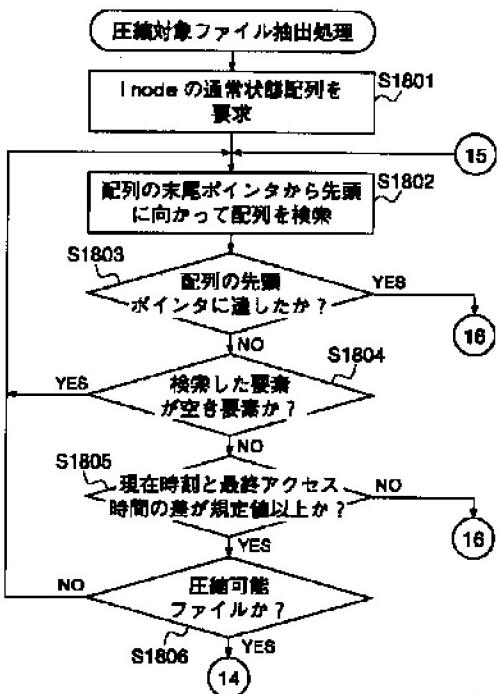
【 図20 】



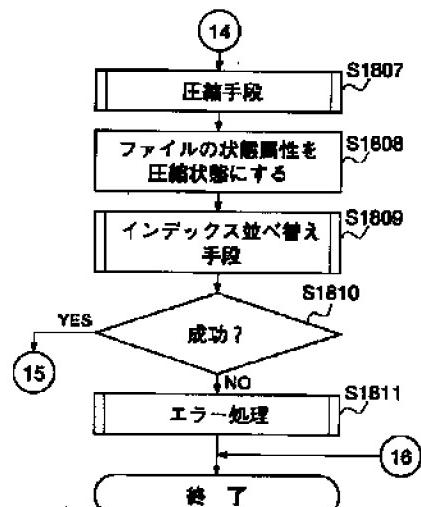
【 図22 】



[図23]



[図24]



フロント ページの続き

(72) 発明者 別所 正隆

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内